

COVER PAGE

Hewlett-Packard Company Docket Number:

10017760-1

Title:

System and Method for Displaying an
Image on a Network Attachable Display Device

Inventors:

Roland M. Hochmuth
2000 Catkins Ct.
Fort Collins, Colorado 80528

Johnny Marks
5212 Madison Creek Drive
Fort Collins, Colorado 80528

10017760-1

SYSTEM AND METHOD FOR
DISPLAYING AN IMAGE ON A NETWORK ATTACHABLE DISPLAY DEVICE

CROSS-REFERENCE TO RELATED APPLICATION

This patent application is related to co-pending U.S. Patent Application, Serial No. _____, entitled "SYSTEM AND METHOD FOR COMMUNICATING GRAPHICS IMAGE DATA OVER A COMMUNICATION NETWORK," filed October 31, 2001, co-assigned herewith.

TECHNICAL FIELD OF THE INVENTION

The present invention relates generally to the field of computer systems, and more particularly to a system and method for displaying an image on a network attachable display device.

BACKGROUND OF THE INVENTION

In existing computer systems, a user's input intended to modify the image on the display is transmitted as a command with the associated data from a CPU to a graphics adapter over an AGP (Accelerated Graphics Port) and a PCI (Peripheral Component Interconnect) bus. A graphics chip on the graphics adapter renders the image based at least in part on the received command. Graphics image data associated with the rendered image is stored in a frame buffer. The image information is typically stored in the frame buffer in a digital format. The digital data stored in the frame buffer controls the color and intensity of every pixel on the display. The graphics adapter may include a RAMDAC (Random Access Memory Digital-to-Analog Converter) to convert graphics image data stored in the frame buffer from digital to analog form for display on an analog display.

卷之三

The RAMDAC may be part of or separate from a display refresh unit. Depending on the refresh rate of the display, the display refresh unit frequently reads the contents of the frame buffer. If desired, the RAMDAC may convert the information into analog form and send it to the monitor. Thus, for a 1280x1024 pixel display using 24 bits per pixel, approximately 4MB of data has to be read from the frame buffer for each refresh of the display. If the refresh rate of the display is 70 times per second, then approximately 280 MB of data is transmitted from the graphics adapter every second.

Two resources in the graphics adapter compete for the same frame buffer - the rendering unit and the display refresh unit. The rendering unit and the display refresh unit may be part of the same graphics chip. Because of the frequent and constant access of the frame buffer by the display refresh unit required to transfer large amounts of data, the rendering performance of the graphics adapter is adversely affected.

Furthermore, it is often desirable to display graphics information that is generated by a computer on a remote display monitor. Sometimes it is desirable to provide this type of remote graphics display across a communication network. One approach to providing graphics image data to a remote display involves reading data from the frame buffer back into the memory of the computer system, compressing the data, assembling the data into IP (Internet Protocol) packets, and then sending the packets to the destination computer over the communication network.

The destination computer receives the compressed image via a network interface card and stores it in system memory. The compressed image is transmitted to a PCI card where it is decompressed. The decompressed image is transmitted back to the system memory. The decompressed image is then transmitted to a graphics adapter on the destination computer via an AGP or a PCI bus. The graphics adapter transmits video signals over a video cable to the display, typically a CRT (Cathode Ray Tube), or LCD (Liquid Crystal Display) to display the graphics image. It may be seen from the foregoing that there is considerable data movement on the destination computer which increases the latency.

SUMMARY OF THE INVENTION

In accordance with an embodiment of the present invention, a network attachable display device comprises a display network interface operable to receive graphics image data of an image over a communication network. The network attachable display device also comprises a display frame buffer operable to store the received graphics image data and a display refresh unit operable to read the stored graphics image data from the display frame buffer. The display refresh unit is further operable to display the image on a display unit.

In accordance with another embodiment of the present invention, a method for displaying an image on a network attachable display device is disclosed. The method comprises receiving, by a display network interface of the network attachable display device, graphics image data of the image over a communication network; storing the received graphics image data in a display frame buffer of the network attachable display device; reading the stored graphics image data from the display frame buffer by a display refresh unit; and displaying the image on a display unit.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention, the objects and advantages thereof, reference is now made to the following descriptions taken in connection with the accompanying drawings in which:

FIGURE 1 is a block diagram of an exemplary system in which a graphics adapter of the present invention may be used;

FIGURES 2A-2B are logical block diagrams of a graphics adapter in accordance with different embodiments of the present invention;

FIGURE 3 is a flowchart describing the operation of a graphics adapter in accordance with an embodiment of the present invention.

FIGURE 4 is a flowchart describing the operation of a graphics adapter in accordance with another embodiment of the present invention;

FIGURE 5 is a flowchart describing the operation of a graphics adapter in accordance with a further embodiment of the present invention;

PCT/US2001/030007

FIGURE 6 is a flowchart describing the operation of a graphics adapter in accordance with yet another embodiment of the present invention;

FIGURES 7A-7B are logical block diagrams of a network attachable display device in accordance with different embodiments of the present invention;

FIGURE 8 is a flowchart describing the operation of a network attachable display device in accordance with an embodiment of the present invention; and

FIGURE 9 is a flowchart describing the operation of a network attachable display device in accordance with another embodiment of the present invention.

DETAILED DESCRIPTION OF THE DRAWINGS

The preferred embodiment of the present invention and its advantages are best understood by referring to FIGURES 1 through 9 of the drawings, like numerals being used for like and corresponding parts of the various drawings.

There is a desire for an improved system and method for communicating graphics image data over a communication network, such as a computer network. Accordingly, a graphics adapter comprising a compression unit and a network interface, capable of communicating with a destination device, such as a remote computer, over a communication network, such as a computer network, is disclosed.

The graphics adapter also includes a memory unit, such as a frame buffer. In addition to or in place of one or more video output ports, such as analog video ports, Digital Visual Interface (DVI) video ports, and the like, that are included in existing graphics adapters, the graphics adapter in accordance with the present invention may be provided with a network interface port, for example an Ethernet port, an Infiniband port, a fiber port, a wireless network transceiver, and the like. The network interface port may be used for transmitting the contents of the frame buffer in either compressed or uncompressed form via the computer network to one or more remote computers.

FIGURE 1 is a block diagram of an exemplary system 10 in which the graphics adapter of the present invention may be used. System 10 preferably comprises at least one source device 12, such as at least one processor-based system, for example computers, and at least one destination device 14. Destination device 14

may comprise, for example a display device capable of being directly coupled to a communication network, a display associated with a computer, and the like. One or more source devices 12 may communicate with each other over a communication network (not shown), such as a local area network. Source device 12 preferably communicates with destination device 14 over a communication network 16. Communication network 16 may comprise an intranet, an extranet and/or the Internet.

Source device 12 may comprise a graphics adapter 20. However, in alternative embodiments, graphics adapter 20 may not be part of source device 12. In such an embodiment, graphics adapter 20 may be coupled directly to communication network 16.

FIGURE 2A is a logical block diagram of a graphics adapter 20 in accordance with an embodiment of the present invention. Graphics adapter 20 comprises a network interface chip 38 coupled to a memory unit 24. A graphics chipset 22 is also preferably coupled to memory unit 24. Network interface chip 38 preferably comprises a compression unit 26 and a network interface 28. If desired, graphics adapter 20 may also comprise at least one video transmitter, for example an analog transmitter 30 and a digital transmitter 32, coupled to memory unit 24. Preferably, analog transmitter 30 and digital transmitter 32 are also coupled to each other. If desired, network interface chip 38 may be coupled to graphics chipset 22.

Graphics chipset 22 is a conventional graphics chipset which renders the image to be displayed on destination device 14. The image may be rendered based at least in part on instructions received from a memory unit of source device 12 or instructions received from destination device 14. Instructions from source device 12 are preferably received via a graphics port 34. Graphics port 34 may be a Peripheral Component Interconnect (PCI) port, an Accelerated Graphics Port (AGP), an Infiniband Port, a fiber port and the like. In an alternative embodiment, instructions may be received via a network interface port 36. In such an embodiment, graphics adapter 20 may not include graphics port 34. Rendering of an image typically comprises translating high-level instruction to bitmap images which are a matrix of pixels. The process of rendering is known in the art and as such will not be described in detail herein.

10017760-1

Memory unit 24 preferably comprises a frame buffer 50. Frame buffer 50 is used to store information about the rendered image. Frame buffer 50 may store information, such as color, transparency values, depth and/or the like, for each pixel in the rendered image.

Compression unit 26 of network interface chip 38 preferably compresses the data stored in frame buffer 50. Compression unit 26 includes compression logic optimized for use with graphics image data stored in frame buffer 50. However, the invention is not so limited and any kind of compression logic may be used by compression unit 26. Compression unit 26 may be implemented using general-purpose hardware components, such as a digital signal processor or microprocessor, and appropriate software for controlling the operation of the hardware components. If desired, compression unit 26 may be implemented using dedicated hardware, an Application Specific Integrated Circuit (ASIC) or a Field Programmable Gate Array (FPGA) that is specially designed for carrying out the compression.

Network interface 28 formats the graphics image data for transmission over a network, such as communication network 16. A network interface port 36, for example an Ethernet port, an Infiniband port, a fiber port, a wireless network transceiver, and the like, is preferably coupled to network interface chip 38. Network interface 28 converts the graphics image data into packets to be transmitted to destination device 14 via network interface port 36 over communication network 16. The graphics image data may be obtained by network interface 28 either from compression unit 26 or from frame buffer 50. For example, in one embodiment, graphics image data from frame buffer 50 may be compressed by compression unit 26 and provided to network interface 28. In another embodiment, after compressing the graphics image data, compression unit 26 may provide the data back to frame buffer 50 and network interface 28 may read the compressed graphics image data directly from frame buffer 50. In yet another embodiment, for example when compression of graphics image data is not desired, network interface 28 may read graphics image data directly from frame buffer 50.

Analog transmitter 30 comprises a conventional RAMDAC and may be used to convert graphics image data stored in memory unit 24 from digital format into an analog format to be transmitted to an analog device, such as an analog monitor which

10017760-100001

may be coupled to source device 12, via analog video port 40. Digital transmitter 32 comprises a conventional Digital Visual Interface (DVI) transmitter and may be used to transmit graphics image data in digital format from memory unit 24 to a digital device, such as a digital monitor which may be coupled to source device 12, via digital video port 42.

FIGURE 2B is a logical block diagram of graphics adapter 20 in accordance with another embodiment of the present invention. Graphics adapter 20 as shown in FIGURE 2B comprises a network attachable graphics chip 52 coupled to memory unit 24. Network attachable graphics chip 52 comprises a graphics unit 44, compression unit 26 and network interface 28. If desired, graphics adapter 20 may also comprise at least one video transmitter, for example analog transmitter 30 and digital transmitter 32, coupled to memory unit 24. Preferably, analog transmitter 30 and digital transmitter 32 are also coupled to each other. In graphics adapter 20 of FIGURE 2B, network interface port 36 is preferably coupled to network attachable graphics chip 52. Graphics unit 44 performs substantially the same function as graphics chipset 22 of FIGURE 2A.

One of the advantages of graphics adapter 20 of FIGURE 2B over the embodiment of FIGURE 2A is that the manufacturing cost of the graphics adapter can be reduced as only one chip, network attachable graphics chip 52, is used in the embodiment of FIGURE 2B.

In an alternative embodiment, graphics adapter 20 does not include analog transmitter 30 or digital transmitter 32. In yet another embodiment, graphics adapter 20 does not include analog video port 40 and digital video port 42; graphics image data is transmitted to source device 12 via network interface port 36.

In yet another embodiment of the present invention, graphics adapter 20 does not include graphics port 34 for receiving graphics commands. Instead graphics adapter 20 receives graphics commands over communication network 16 via network interface port 36. The graphics command may be received from source device 12 or from destination device 14.

FIGURE 3 is a flowchart 90 describing the operation of graphics adapter 20 in accordance with an embodiment of the present invention. In step 92, the graphics adapter waits for either the lapse of a predetermined period of time or the receipt of an

10017760-1
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
25100
25101
25102
25103
25104
25105
25106
25107
25108
25109
25110
25111
25112
25113
25114
25115
25116
25117
25118
25119
25120
25121
25122
25123
25124
25125
25126
25127
25128
25129
25130
25131
25132
25133
25134
25135
25136
25137
25138
25139
25140
25141
25142
25143
25144
25145
25146
25147
25148
25149
25150
25151
25152
25153
25154
25155
25156
25157
25158
25159
25160
25161
25162
25163
25164
25165
25166
25167
25168
25169
25170
25171
25172
25173
25174
25175
25176
25177
25178
25179
25180
25181
25182
25183
25184
25185
25186
25187
25188
25189
25190
25191
25192
25193
25194
25195
25196
25197
25198
25199
25200
25201
25202
25203
25204
25205
25206
25207
25208
25209
25210
25211
25212
25213
25214
25215
25216
25217
25218
25219
25220
25221
25222
25223
25224
25225
25226
25227
25228
25229
25230
25231
25232
25233
25234
25235
25236
25237
25238
25239
25240
25241
25242
25243
25244
25245
25246
25247
25248
25249
25250
25251
25252
25253
25254
25255
25256
25257
25258
25259
25260
25261
25262
25263
25264
25265
25266
25267
25268
25269
25270
25271
25272
25273
25274
25275
25276
25277
25278
25279
25280
25281
25282
25283
25284
25285
25286
25287
25288
25289
25290
25291
25292
25293
25294
25295
25296
25297
25298
25299
252100
252101
252102
252103
252104
252105
252106
252107
252108
252109
252110
252111
252112
252113
252114
252115
252116
252117
252118
252119
252120
252121
252122
252123
252124
252125
252126
252127
252128
252129
252130
252131
252132
252133
252134
252135
252136
252137
252138
252139
252140
252141
252142
252143
252144
252145
252146
252147
252148
252149
252150
252151
252152
252153
252154
252155
252156
252157
252158
252159
252160
252161
252162
252163
252164
252165
252166
252167
252168
252169
252170
252171
252172
252173
252174
252175
252176
252177
252178
252179
252180
252181
252182
252183
252184
252185
252186
252187
252188
252189
252190
252191
252192
252193
252194
252195
252196
252197
252198
252199
252200
252201
252202
252203
252204
252205
252206
252207
252208
252209
252210
252211
252212
252213
252214
252215
252216
252217
252218
252219
252220
252221
252222
252223
252224
252225
252226
252227
252228
252229
252230
252231
252232
252233
252234
252235
252236
252237
252238
252239
252240
252241
252242
252243
252244
252245
252246
252247
252248
252249
252250
252251
252252
252253
252254
252255
252256
252257
252258
252259
252260
252261
252262
252263
252264
252265
252266
252267
252268
252269
252270
252271
252272
252273
252274
252275
252276
252277
252278
252279
252280
252281
252282
252283
252284
252285
252286
252287
252288
252289
252290
252291
252292
252293
252294
252295
252296
252297
252298
252299
252300
252301
252302
252303
252304
252305
252306
252307
252308
252309
252310
252311
252312
252313
252314
252315
252316
252317
252318
252319
252320
252321
252322
252323
252324
252325
252326
252327
252328
252329
252330
252331
252332
252333
252334
252335
252336
252337
252338
252339
252340
252341
252342
252343
252344
252345
252346
252347
252348
252349
252350
252351
252352
252353
252354
252355
252356
252357
252358
252359
252360
252361
252362
252363
252364
252365
252366
252367
252368
252369
252370
252371
252372
252373
252374
252375
252376
252377
252378
252379
252380
252381
252382
252383
252384
252385
252386
252387
252388
252389
252390
252391
252392
252393
252394
252395
252396
252397
252398
252399
252400
252401
252402
252403
252404
252405
252406
252407
252408
252409
252410
252411
252412
252413
252414
252415
252416
252417
252418
252419
252420
252421
252422
252423
252424
252425
252426
252427
252428
252429
252430
252431
252432
252433
252434
252435
252436
252437
252438
252439
252440
252441
252442
252443
252444
252445
252446
252447
252448
252449
252450
252451
252452
252453
252454
252455
252456
252457
252458
252459
252460
252461
252462
252463
252464
252465
252466
252467
252468
252469
252470
252471
252472
252473
252474
252475
252476
252477
252478
252479
252480
252481
252482
252483
252484
252485
252486
252487
252488
252489
252490
252491
252492
252493
252494
252495
252496
252497
252498
252499
252500
252501
252502
252503
252504
252505
252506
252507
252508
252509
252510
252511
252512
252513
252514
252515
252516
252517
252518
252519
252520
252521
252522
252523
252524
252525
252526
252527
252528
252529
252530
252531
252532
252533
252534
252535
252536
252537
252538
252539
252540
252541
252542
252543
252544
252545
252546
252547
252548
252549
252550
252551
252552
252553
252554
252555
252556
252557
252558
252559
252560
252561
252562
252563
252564
252565
252566
252567
252568
252569
252570
252571
252572
252573
252574
252575
252576
252577
252578
252579
252580
252581
252582
252583
252584
252585
252586
252587
252588
252589
252590
252591
252592
252593
252594
252595
252596
252597
252598
252599
2525100
2525101
2525102
2525103
2525104
2525105
2525106
2525107
2525108
2525109
2525110
2525111
2525112
2525113
2525114
2525115
2525116
2525117
2525118
2525119
2525120
2525121
2525122
2525123
2525124
2525125
2525126
2525127
2525128
2525129
2525130
2525131
2525132
2525133
2525134
2525135
2525136
2525137
2525138
2525139
2525140
2525141
2525142
2525143
2525144
2525145
2525146
2525147
2525148
2525149
2525150
2525151
2525152
2525153
2525154
2525155
2525156
2525157
2525158
2525159
2525160
2525161
2525162
2525163
2525164
2525165
2525166
2525167
2525168
2525169
2525170
2525171
2525172
2525173
2525174
2525175
2525176
2525177
2525178
2525179
2525180
2525181
2525182
2525183
2525184
2525185
2525186
2525187
2525188
2525189
2525190
2525191
2525192
2525193
2525194
2525195
2525196
2525197
2525198
2525199
2525200
2525201
2525202
2525203
2525204
2525205
2525206
2525207
2525208
2525209
2525210
2525211
2525212
2525213
2525214
2525215
2525216
2525217
2525218
2525219
2525220
2525221
2525222
2525223
2525224
2525225
2525226
2525227
2525228
2525229
2525230
2525231
2525232
2525233
2525234
2525235
2525236
2525237
2525238
2525239
2525240
2525241
2525242
2525243
2525244
2525245
2525246
2525247
2525248
2525249
2525250
2525251
2525252
2525253
2525254
2525255
2525256
2525257
2525258
2525259
25252510
25252511
25252512
25252513
25252514
25252515
25252516
25252517
25252518
25252519
25252520
25252521
25252522
25252523
25252524
25252525
25252526
25252527
25252528
25252529
252525210
252525211
252525212
252525213
252525214
252525215
252525216
252525217
252525218
252525219
252525220
252525221
252525222
252525223
252525224
252525225
252525226
252525227
252525228
252525229
252525230
252525231
252525232
252525233
252525234
252525235
252525236
252525237
252525238
252525239
252525240
252525241
252525242
252525243
252525244
252525245
252525246
252525247
252525248
252525249
252525250
252525251
252525252
252525253
252525254
252525255
252525256
252525257
252525258
252525259
252525260
252525261
252525262
252525263
252525264
252525265
252525266
252525267
252525268
252525269
252525270
252525271
252525272
252525273
252525274
252525275
252525276
252525277
252525278
252525279
252525280
252525281
252525282
252525283
252525284
252525285
252525286
252525287
252525288
252525289
252525290
252525291
252525292
252525293
252525294
252525295
252525296
252525297
252525298
252525299
2525252100
2525252101
2525252102
2525252103
2525252104
2525252105
2525252106
2525252107
2525252108
2525252109
2525252110
2525252111
2525252112
2525252113
2525252114
2525252115
2525252116
2525252117
2525252118
2525252119
2525252120
2525252121
2525252122
2525252123
2525252124
2525252125
2525252126
2525252127
2525252128
2525252129
2525252130
2525252131
2525252132
2525252133
2525252134
2525252135
2525252136
2525252137
2525252138
2525252139
2525252140
2525252141
2525252142
2525252143
2525252144
2525252145
2525252146
2525252147
2525252148
2525252149
2525252150
2525252151
2525252152
2525252153
2525252154
2525252155
2525252156
2525252157
2525252158
2525252159
2525252160
2525252161
2525252162
2525252163
2525252164
2525252165
2525252166
2525252167
2525252168
2525252169
2525252170
2525252171
2525252172
2525252173
2525252174
2525252175
2525252176
2525252177
2525252178
2525252179
2525252180
2525252181
2525252182
2525252183
2525252184
2525252185
2525252186
2525252187
2525252188
2525252189
2525252190
2525252191
2525252192
2525252193
2525252194
2525252195
2525252196
2525252197
2525252198
2525252199
2525252200
2525252201
2525252202
2525252203
2525252204
2525252205
2525252206
2525252207
2525252208
2525252209
2525252210
2525252211
2525252212
2525252213
2525252214
2525252215
2525252216
2525252217
2525252218
2525252219
2525252220
2525252221
2525252222
2525252223
2525252224
2525252225
2525252226
2525252227
2525252228
2525252229
2525252230
2525252231
2525252232
2525252233
2525252234
2525252235
2525252236
2525252237
2525252238
2525252239
2525252240
2525252241
2525252242
2525252243
2525252244
2525252245
2525252246
2525252247
2525252248
25

update request. The predetermined period of time may be dependent upon the refresh rate of a display associated with destination device 14. Destination device 14 itself may be the display. The update request may be received by graphics adapter 20 from source device 12 via graphics port 34. The request may also be received by the graphics adapter from destination device 14 via network interface port 36 over communication network 16.

Upon the occurrence of one of the above events, a determination is made in step 94 as to whether the graphics image data in frame buffer 50 has been changed since the last update. If the graphics image data in frame buffer 50 has not been changed, then the process starting at step 92 is repeated. The step of determining whether the graphics image data in frame buffer 50 has been changed is optional and has been shown in FIGURE 3 by dashed lines.

In the preferred embodiment of step 94, a flag is read to determine whether the graphics image data in frame buffer 50 associated with the destination device has been updated since the last transmission. The flag may have been previously set by graphics chipset 22 during a write to frame buffer 50. For example, when information for each pixel is written into frame buffer 50, the pixel value could be compared with a previously stored value, and the flag is set if there has been a change. If desired, the particular region of frame buffer 50 associated with the destination device that has changed may also be tracked, for example by tracking the minimum and maximum values in the x and y coordinates for the region that has changed.

In step 98, graphics image data from frame buffer 50 is read by compression unit 26. If desired, color coalescing may be performed in this step. A Display Refresh Unit (DRU) may perform the task of color coalescing. The DRU may be part of the network interface chip or may be coupled to the network interface chip. Color coalescing is the process of merging images of multiple display surfaces, such as a primary display surface and one or more overlay planes into a single image. One or more color look-up tables containing colors for the pixels may be stored in registers (not shown) on graphics chipset 22. A window identifier may be used to determine the color look-up table to be used. Based on the value for the pixels stored in frame buffer 50, the colors for the pixels are obtained from the determined color look-up table.

In step 100, a determination is made as to whether compression is enabled. A compression bit within a register (not shown) on graphics chipset 22 may be read to determine if compression is enabled or not. If compression is enabled, then in step 102, graphics image data read from frame buffer 50 is compressed preferably by compression unit 26. Preferably, a compression algorithm optimized for compressing graphics image data read from frame buffer 50 is used. However, any of a variety of compression algorithms may be used by compression unit 26 to compress the data read from frame buffer 50. After compression, compression unit 26 may write the compressed data back to frame buffer 50. Compression of graphics image data prior to transmission over communication network 16 significantly reduces the amount of data that is transmitted over communication network 16.

If, in step 100, it is determined that compression is not enabled, then execution proceeds directly to step 104. In step 104, graphics image data is formatted for communication over communication network 16, preferably by network interface 28. Network interface 28 may read graphics image data from frame buffer 50 or may receive graphics image data directly from compression unit 26. In a preferred embodiment, graphics image data, whether in compressed form or in uncompressed form, is formatted into a plurality of packets suitable for transmission over communication network 16 using a communication protocol, such as Internet Protocol (IP). The formatted data packets include, among other things, identification information of destination device 14 to which the graphics image data is to be transmitted. Identification information may include for example an IP address of destination device 14. Identification information about destination device 14 may be received by graphics adapter 20 either from source device 12 or from destination device 14 itself.

In step 106, the formatted graphics image data is transmitted over communication network 16 to destination device 14, preferably as one or more Internet Protocol (IP) packets. The packets include a portion of the graphics image data (payload) and identification information about destination device 14. If desired, the same graphics image data may be transmitted to more than one destination device 14. This may be accomplished, for example, by creating data packets with the same payload but different identification information corresponding to the different

10017760-1

destination devices. If desired, the formatted graphics image data may be transmitted via graphics port 34 to source device 12. Transmitting the graphics image data to source device 12 allows the source device to perform further processing of the graphics image data as desired. By transmitting packets with the same payload to more than one destination device, the same data may be presented to users of multiple destination devices, thereby enabling sharing of graphics image data by these multiple users. If desired, different graphics image data may be transmitted from graphics adapter 20 to the different users as described herein with reference to FIGURE 6.

FIGURE 4 is a flowchart 110 describing the operation of a graphics adapter in accordance with another embodiment of the present invention. In step 112, the graphics adapter receives a request for an updated image from destination device 14. The request is received preferably over communication network 16 via network interface port 36. Information identifying destination device 14 may also be received from destination device 14.

In step 116, graphics image data is stored in frame buffer 50. In step 118, graphics image data from frame buffer 50 is read preferably by compression unit 26. If desired, color coalescing may be performed in this step. In step 120, a determination is made as to whether compression is enabled. If compression is enabled, then in step 122, graphics image data read from frame buffer 50 is compressed preferably by compression unit 26.

If, in step 120, it is determined that compression is not enabled, then execution proceeds directly to step 124. In step 124, the graphics image data is formatted for communication over communication network 16, preferably by network interface 28. Network interface 28 may read graphics image data from frame buffer 50 or may receive graphics image data directly from compression unit 26. In a preferred embodiment, graphics image data, whether in compressed form or in uncompressed form, is formatted into a plurality of packets suitable for transmission over communication network 16 using a communication protocol, such as Internet Protocol (IP). The formatted data packets include, among other things, identification information of destination device 14 to which the graphics image data is to be transmitted. Identification information may include for example an IP address of destination device 14. In step 126, the formatted graphics image data is transmitted

10017760-1

over communication network 16 to destination device 14, preferably as one or more IP packets.

FIGURE 5 is a flowchart 130 describing the operation of a graphics adapter in accordance with a further embodiment of the present invention. In the method of flowchart 130 only the portions of the frame buffer that have changed since the last update are transmitted to destination device 14. Graphics image data for a display may be effectively partitioned into a plurality of blocks, for example 16,384 blocks. As mapped to a display, these blocks may comprise an array of 128 blocks by 128 blocks. In a display having a resolution of 1,280 pixels by 1,024 pixels, each block has a corresponding resolution of 10 pixels by 8 pixels. However, the present invention is not so limited and a number of different block sizes may be chosen based upon system resources, the nature of the graphics image data to be transmitted, and a number of other factors.

Preferably, frame buffer 50 of graphics adapter of FIGURES 2A-2B stores information about both a new image and a previous image so that comparisons between the two images may be made. In step 132, the graphics adapter waits for either the lapse of a predetermined period of time or the receipt of an update request. The predetermined period of time may be dependent upon the refresh rate of a display associated with destination device 14. The request for an update may be received from source device 12 with which the graphics adapter is associated via graphics port 34. The request may be received by the graphics adapter from destination device 14 via network interface port 36 over communication network 16.

In step 134, portions or blocks of the graphics image data stored in the frame buffer corresponding to the new image are compared to corresponding portions or blocks of the graphics image data stored in the frame buffer corresponding to the previous image. In step 136, a determination is made as to whether any portions have changed since the last update. If the graphics image data has not been changed then the process starting at step 132 is repeated.

In step 138, a determination is made as to whether compression is enabled. If compression is enabled, then in step 140, graphics image data for portions or blocks of the new image that are determined to be different from the corresponding portions or blocks of the previous image are compressed. In an alternative embodiment, only

10017760-1

those blocks that have changed since the last update and are within a "region of interest" specified by destination device 14 are compressed. This may be desirable, for example, when destination device 14 only desires to update specific portions of its display, for example a specific window of a plurality of windows.

5 If, at step 138, it is determined that compression is not enabled, then execution proceeds directly to step 142. In step 142, graphics image data for blocks that have changed is formatted for communication over communication network 16, preferably by network interface 28. In a preferred embodiment, graphics image data in any block with any change at all from the previously stored block may be formatted. If
10 desired, some threshold for change may be set. The graphics adapter may be configured to evaluate the degree or amount of change and format the graphics image data only if it has changed by some predetermined amount.

15 Network interface 28 may read graphics image data from frame buffer 50 or may receive graphics image data directly from compression unit 26. In a preferred embodiment, graphics image data associated with any blocks that have changed, whether in compressed form or in uncompressed form, is formatted into a plurality of packets suitable for transmission over communication network 16 using a communication protocol, such as Internet Protocol (IP). A packet may include one or more blocks of graphics image data. The formatted data packets include, among other things, identification information of destination device 14 to which the graphics image data is to be transmitted. Identification information may include for example an IP address of destination device 14. Identification information about destination device 14 may be provided to the graphics adapter either from source device 12 or from destination device 14 itself.

20 25 In step 144, block identifiers (IDs) are added to the packets of data to be transmitted. The block ID information is preferably used by destination device 14 for updating a display associated with it. A block ID identifies the particular block for which graphics image data is being transmitted. A block ID may be a block number. If desired, the minimum and maximum values in the x and y coordinates (X_{min} , X_{max} ,
30 Y_{min} , Y_{max}) for the block may be used as the block ID. The four corners of the block that is identified by the minimum and maximum values in the x and y coordinates are given by (X_{min} , Y_{min}), (X_{max} , Y_{min}), (X_{max} , Y_{max}) and (X_{min} , Y_{max}).

In step 146, formatted graphics image data is transmitted over communication network 16 to destination device 14, preferably as one or more IP packets. If, desired, formatted graphics image data may be transmitted over graphics port 34 to source device 12. The same graphics image data may be transmitted to one or more destination devices 14. This may be accomplished, for example, by creating data packets with the same payload but including identification information, such as IP addresses, for different destination devices in the data packets.

In step 148, one or more blocks of frame buffer 50 corresponding to the previous image are updated. Preferably, only the blocks of the frame buffer associated with the previous image that have changed are overwritten by graphics image data for the corresponding blocks of the new image. If desired, the entire portion of the frame buffer associated with the previous image may be overwritten by graphics image data from the frame buffer associated with the new image.

It may be desirable to periodically transmit an entire frame buffer associated with a display to avoid problems that may occur due to packets lost in transmission or otherwise. If desired, destination device 14 may be configured to transmit acknowledgements back to the graphics adapter to better ensure proper synchronization between the graphics adapter and destination device 14.

FIGURE 6 is a flowchart 150 describing the operation of a graphics adapter in accordance with yet another embodiment of the present invention. The method as described herein with reference to FIGURE 6 may be used to provide support for multiple destination devices by the same graphics adapter. The graphics adapter of the present invention may be used to render graphics for multiple destination devices, such as multiple computers, coupled to communication network 16. Graphics image data may be transmitted from the graphics adapter to the destination devices via network interface port 36 over communication network 16 without the need for separate ports for the different destination devices. Thus, the number of destination devices that may be supported is not limited by the number of physical ports associated with the graphics adapter.

In step 152, the graphics adapter waits for either the lapse of a predetermined period of time or the receipt of an update request. The predetermined period of time may be dependent upon the refresh rate of a display associated with a destination

device. The request for an update may be received by the graphics adapter from source device 12 with which the graphics adapter is associated via graphics port 34. The request may be received by the graphics adapter from destination device 14 via network interface port 36 over communication network 16.

Upon the occurrence of one of the above events, in step 154, the portion of frame buffer 50 corresponding to destination device 14 to which graphics image data is to be transmitted is determined. In the preferred embodiment, a destination device table may be looked up to determine the portion of the frame buffer 50 corresponding to the destination device. The destination device table preferably includes information about one or more destination devices, such as identification information of the destination device, the region of frame buffer 50 assigned to the destination device, information indicating whether graphics image data corresponding to the destination device has changed, and/or the like.

In step 156 a determination is made as to whether the graphics image data corresponding to the destination device in frame buffer 50 has been changed since the last update. If the graphics image data in frame buffer 50 has not been changed, then the process starting at step 152 is repeated. The step of determining whether the graphics image data in frame buffer 50 has been changed is optional and has been shown in FIGURE 6 by dashed lines.

In step 158, a determination is made as to whether compression is enabled. If compression is enabled, then in step 160, graphics image data stored in the portion of frame buffer 50 associated with the particular destination device is compressed preferably by compression unit 26.

If, in step 158, it is determined that compression is not enabled, then execution proceeds directly to step 162. In step 162, the graphics image data associated with the particular destination device 14 is formatted for communication over communication network 16, preferably by network interface 28. Network interface 28 may read graphics image data from frame buffer 50 or may receive graphics image data directly from compression unit 26. In a preferred embodiment, graphics image data, whether in compressed form or in uncompressed form, is formatted into a plurality of packets suitable for transmission over communication network 16 using a communication protocol, such as Internet Protocol. The formatted data packets include, among other

things, identification information of destination device 14 to which the graphics image data is to be transmitted. Identification information may include for example an IP address of the particular destination device 14. Identification information about destination device 14 may be received by the graphics adapter either from source device 12 or from destination device 14 itself.

5 In step 164, the formatted graphics image data is transmitted over communication network 16 to destination device 14, preferably as one or more IP packets.

A technical advantage of an exemplary embodiment of the graphics adapter of
10 the present invention is that graphics image data does not have to be transferred back into the memory of the source computer in order to be transferred to a destination device. Because the graphics adapter may be directly connected to a communication network, graphics image data from the graphics adapter may be directly transmitted over the communication network to the destination device. Another technical advantage of an exemplary embodiment of the graphics adapter of the present invention is that graphics image data of the frame buffer is transmitted to the destination device only if the contents of the frame buffer have changed. Moreover, if desired, graphics image data for only particular regions of interest, for example, the blocks of the frame buffer that have changed, may be transmitted, thereby eliminating
15 the need to transmit the contents of the entire frame buffer with every update.

Moreover, the graphics image data may be tagged or addressed and sent to more than one destination device. Thus, multiple destination devices may be simultaneously supported by the same graphics adapter without being limited by the number of physical ports associated with the graphics adapter. The multiple destination devices may be provided with the same image data thereby allowing multiple users to collaborate on the same project. If desired, different portions of the frame buffer may be allocated to different destination devices and the multiple destination devices provided with different data.

25 FIGURE 7A is a logical block diagram of a network attachable display device 14 in accordance with an embodiment of the present invention. Network attachable display device 14 comprises a display network interface 172 coupled to a display memory unit 174, a display decompression unit 176 and a display refresh unit 178.

10017760-1

Display memory unit 174 is also coupled to display decompression unit 176 and display refresh unit 178. Display decompression unit 176 is also coupled to display refresh unit 178 and display refresh unit 178 is also coupled to a display unit 180.

Network attachable display device 14 of the preferred embodiment is capable of being coupled to a communication network, such as communication network 16. Thus, network attachable display device 14 may communicate with other devices coupled to communication network 16 and may receive and/or transmit commands and/or data to and from such devices over communication network 16 via network interface port 182.

Display network interface 172 is operable to receive data, such as graphics image data, from a source device, such as source device 12, graphics adapter 20, a graphics appliance, and/or the like, over a communication network, such as communication network 16, via a display network interface port 182, for example an Ethernet port, an Infiniband port, a fiber port, a wireless network transceiver, and the like. The graphics image data is preferably received in a packetized format.

Display memory unit 174 preferably comprises a display frame buffer 170. Display frame buffer 170 is used to store graphics image data that is to be displayed on display unit 180. The graphics image data may be stored in display frame buffer 170 in either compressed or decompressed form. Display frame buffer 170 may store information, such as color for each pixel to be displayed on display unit 180.

Display decompression unit 176 preferably decompresses the data received by display network interface 172. Display decompression unit 176 includes decompression logic optimized for use with graphics image data. However, the invention is not so limited and any kind of decompression logic may be used by display decompression unit 176. Display decompression unit 176 may be implemented using general-purpose hardware components, such as a digital signal processor or microprocessor, and appropriate software for controlling the operation of the hardware components. If desired, display decompression unit 176 may be implemented using dedicated hardware, an ASIC or a FPGA that is specially designed for carrying out the decompression. In the embodiment shown in FIGURE 7A, display decompression unit 176 is also coupled to display network interface 172 and display refresh unit 178.

1000177601 1000177602 1000177603 1000177604 1000177605

Display refresh unit 178 is a conventional display refresh unit typically used in graphics adapters. In the embodiment shown in FIGURE 7A, display refresh unit 178 is coupled to display unit 180. Display unit 180 preferably comprises a CRT (Cathode Ray Tube). In alternative embodiments, display unit 180 may comprise a 5 Liquid Crystal Display (LCD), a Thin Film Transistor (TFT), a Light Emitting Diode (LED), organic polymers and/or the like now known or later developed. Depending on the display refresh rate of display unit 180, display refresh unit 178 reads the graphics image data from display frame buffer 170 frequently and refreshes display unit 180 at the refresh rate of display unit 180.

10 Display unit 180 is preferably logically divided into horizontal segments. Preferably display unit 180 is refreshed starting from the top left corner of the desired portion of display unit 180 and moving to the right. Once a horizontal segment of display unit 180 is refreshed, the next horizontal segment is refreshed starting from the left and moving to the right until the desired portion of display unit 180 is refreshed. 15 This process is repeated frequently depending on the display refresh rate of display unit 180.

FIGURE 7B is a logical block diagram of a network attachable display device 14 in accordance with another embodiment of the present invention. Network attachable display device 14 as shown in FIGURE 7B comprises a network attachable display controller 186 coupled to display memory unit 174 and display unit 180. Network attachable display controller 186 comprises display network interface 172, display decompression unit 176 and display refresh unit 178. In network attachable display device 14 of FIGURE 7B, display network interface port 182 is coupled to network attachable display controller 186.

20 One of the advantages of network attachable display device 14 of FIGURE 7B over network attachable display device 14 of FIGURE 7A is that the manufacturing cost of the network attachable display device can be reduced as a single chip, network attachable display controller 186, includes the functionality of display network interface 172, display decompression unit 176 and display refresh unit 178.

25 FIGURE 8 is a flowchart 190 of a method of operation of a network attachable display device in accordance with an embodiment of the present invention. In step 192, graphics image data is received from a source device, such as a computer, a

TOP SECRET - SECURITY INFORMATION

graphics device, a server, an Internet appliance and/or the like, for example source device 12 or graphics adapter 20, over a communication network, such as communication network 16, via display network interface port 182. The data is received preferably as a plurality of packets. Display network interface 172 assembles the received data in a format suitable for storage in display frame buffer 170. For example, if the packets of data include information indicating the portions of display unit 180 with which the data included in the packets are associated, then display network interface 172 stores the data in the appropriate portions of display frame buffer 170.

In step 194, the assembled graphics image data is stored in display frame buffer 170. In step 196, a determination is made as to whether the graphics image data stored in display frame buffer 170 is in compressed form. If the stored graphics image data is in compressed form, then in step 198 the stored graphics image data is decompressed, preferably by display decompression unit 176. In step 200, decompressed data is stored in display frame buffer 170.

If, in step 196, it is determined that the graphics image data stored in display frame buffer 170 is not in compressed form, then execution proceeds directly to step 202. In step 202, graphics image data is read from display frame buffer 170. Preferably, display refresh unit 178 reads the data from display frame buffer 170 and refreshes display unit 180 at the display refresh rate of display unit 180.

FIGURE 9 is a flowchart 206 of a method of operation of a network attachable display device in accordance with another embodiment of the present invention. In step 208, graphics image data is received from a source device, such as a computer, a graphics device, a server, an Internet appliance and/or the like, for example source device 12 or graphics adapter 20, over a communication network, such as communication network 16 via display network interface port 182. The data is received preferably as a plurality of packets.

In step 210, a determination is made as to whether the graphics image data stored in display frame buffer 170 is in compressed form. If the graphics image data is in compressed form, then in step 212, display network interface 172 provides the graphics image data to display decompression unit 176 and the graphics image data is decompressed, preferably by display decompression unit 176, preferably “on the fly”.

10017760-1

For this purpose graphics image data received by display network interface 172 is transferred to display decompression unit 176 and display decompression unit 176 decompresses the graphics image data “on the fly”.

5 If, in step 210, it is determined that the graphics image data stored in display frame buffer 170 is not in compressed form, then execution proceeds directly to step 214. In step 214, decompressed data is stored in display frame buffer 170.

In step 216, graphics image data is read from display frame buffer 170. Preferably, display refresh unit 178 reads the data from display frame buffer 170 and refreshes display unit 180 at the display refresh rate of display unit 180.

10 Preferably, display frame buffer 170 stores graphics image data corresponding to both a new image and a previous image. Graphics image data corresponding to the previous image is preferably graphics image data for the image currently displayed on display unit 180. Graphics image data corresponding to the new image is preferably graphics image data for the image to be displayed on display unit 180. Once display decompression unit 176 or display network interface 172 has stored graphics image data corresponding to the new image in display frame buffer 170, display refresh unit 178 may read graphics image data from the portion of display frame buffer 170 corresponding to the new image. After a new image has been displayed on display unit 180, the graphics image data corresponding to the previous image may be 15 overwritten by the graphics image data for the new image. Thereafter, display refresh unit 178 may read graphics image data from the portion of display frame buffer 170 corresponding to the previous image and refresh display unit 180 at the display refresh rate.

20 The use of display frame buffer 170 to store graphics image data related to both the new image and the previous image prevents “tearing” of the displayed image which may otherwise result if the display refresh unit 178 is refreshing the display unit 180 at a rate which is faster than the rate at which graphics image data related to the new image is being written to display frame buffer 170. The use of display frame buffer 170 to store graphics image data related to both the new image and the previous image may be avoided if graphics image data corresponding to the new image is stored in display frame buffer 170 at least as fast as the rate at which display 25 unit 180 is refreshed.

A technical advantage of an exemplary embodiment of the network attachable display device of the present invention is that graphics image data does not have to be transferred into the memory of a computer in order to be displayed on the network attachable display device. Because the network attachable display device may be directly connected to a communication network, graphics image data may be directly received over the communication network by the network attachable display device itself.

